SAVONIA

OPINNÄYTETYÖ - **AMMATTIKORKEAKOULUTUTKINTO**
TEKNIIKAN JA LIIKENTEEN ALA

MPLS – VERKON QOS-JONOJEN MITTAUS JA TARKASTELU

TEKIJÄ/T: Kallepekka Lumpo

SAVONIA-AMMATTIKORKEAKOULU

OPINNÄYTETYÖ
Tiivistelmä

Koulutusala Tekniikan ja liikenteen ala	
Koulutusohjelma/Tutkinto-ohjelma Tietotekniikan koulutusohjelma	
Työn tekijä(t) Kallepekka Lumpo	
Työn nimi MPLS-verkon QoS-jonojen mittaus ja tarkastelu	
Päiväys 7.11.2017	Sivumäärä/Liitteet 30
Ohjaaja(t) Pekka Vedenpää, Laboratorioinsinööri	
Toimeksiantaja/Yhteistyökumppani(t) Istekki Oy, Mikko Väätäinen	

Tiivistelmä

Työn aiheena oli Istekki Oy: n MPLS-runkoverkon QoS-jonojen monitorointikeinojen parantaminen. Tarkoituksena oli kehitellä menetelmä, jolla QoS-jonoja pystyttäisiin mittaamaan sekä seuraamaan Istekin tarpeiden mukaisesti. Istekillä ei alun perin ollut käytössä monitorointimenetelmää, mutta muutamia ehdotuksia siihen liittyen löytyi.

Työn alussa tehtiin paljon tutkimus- ja tutustumistyötä liittyen työhön. Tässä vaiheessa päätettiin, että käytetään Torrus-nimistä ohjelmaa monitorointiin. Juniperin osalta perehdyttiin sen reitittimiin ja kytkimiin, sekä verkoitus-tekniikoihin, kun taas Torruksen osalta yleisesti vain palveluun ja sen konfigurointiin. Tutkimus- ja tutustumistyö antoi hyvän pohjan opinnäytetyössä tehtyyn käytännön osuuteen, jonka avulla monitorointi voitiin toteuttaa.

Käytännön osuudessa rakennettiin testilaboratorioverkko Savonian tiloihin, jotta voitiin testata QoS-monitorointi käytännössä. Kartoitettiin myös tarkasti, mitä vaatimuksia testiverkossa täytyy olla, jotta monitoroinnista saatiin Istekin tarpeiden mukainen.

Monitorointi ja testiverkon rakennus sujuivat suunnitelmien mukaan. Tuloksia saatiin ja Torrukseen saatiin kehitysideoita, jotka auttoivat Istekin monitorointia eteenpäin. Työtä tullaan jatkossa kehittämään lisää Istekin sisäisesti, sillä Torrus on vielä kehittyvä palvelu.

Avainsanat
MPLS, QOS

SAVONIA UNIVERSITY OF APPLIED SCIENCES

THESIS
Abstract

Field of Study Technology, Communication and Transport			
Degree Programme Degree Programme in Information Technology			
Author(s) Kallepekka Lumpo			
Title of Thesis Measurement and Inspection of QOS Queue in the MPLS Network			
Date	7 November 2017	Pages/Appendices	30
Supervisor(s) Mr. Pekka Vedenpää, Laboratory Engineer			
Client Organisation /Partners Istekki Oy, Mr. Mikko Väätäinen			

Abstract

The subject of the thesis was the improvement of the monitoring of QoS queues for the MPLS backbone network of Istekki Oy. The aim was to develop a method for measuring QoS queues according to Istekki's needs. Istekki initially did not use the monitoring method, but there were a few suggestions regarding it.

At the beginning of the work a lot of research and familiarization work was done. At this point, it was decided to use the program named Torrus for monitoring development. For Juniper its routers and switches as well as in networking techniques were studied while with Torrus generally only service and its configuration about Torrus were studied. The research and familiarization provided a good basis for the practical part of the thesis, which enabled the things studied to be realized.

In the practical part, a test laboratory network was set up in Savonia's premises to test QoS monitoring in practice. It was also mapped out exactly what requirements the test network must have to equivalent to the monitoring at Istekki.

The monitoring and setting up of the test network went as planned. Results were obtained and as well as development ideas that helped Istekki's monitoring forward. The monitoring work will continue to be further developed within Istekki as Torrus is still a developing service.

Keywords
MPLS, QOS, SNMP

ESIPUHE

Haluan kiittää Istekki Oy:tä ja erityisesti Mikko Väätäistä mahdollisuudesta tehdä tämä työ, sekä lisäksi kiittää erinomaisesta ohjauksesta opinnäytetyön kanssa. Kiitokset kuuluvat myös Pekka Vedenpäälle tiloista ja laitteista Savonialla.

Kuopiossa 7.11.2017
Kallepekka Lumpo

SISÄLLYS

LYHENTEET JA MÄÄRITELMÄT

MPLS = Multiprotocol Label Switching

QoS = Quality of Service (Palvelun laatu)

CoS = Class of Service

SNMP = Simple Network Management Protocol

Label = Tunniste

Label stack = Tunnistepino

LSR = Label Switching Router

LSP = Label Switched Path (Tunnistepolku)

FEC = Forwarding Equivalance Class

LDP = Label Distiribution Protocol

LFIB = Label Forwarding Information Base

VPN = Virtual Private Network

Topologia = Kuvaus verkosta ja sen rakenteesta

PE = Provider Edge, Palveluntarjoajan reuna

CE = Customer Edge, Asiakkaan reuna

PyEZ = Junosin yksi kirjasto

WRR = Weighted Round Robin

RED = Random Early Detection

WRED = Weighted Random Early Detection

WFQ = Weighted Fair Queuing

MIC = Modulaarinen liitäntäkortti

PoE = Power over Ethernet

IP = Internet Protocol

1 JOHDANTO

Työssä kehitetään Istekki Oy: n MPLS-runkoverkon QoS-jonojen monitorointikeinoja. Tarkoituksena oli kehitellä menetelmä, jolla QoS-jonoja pystyttäisiin mittaamaan sekä seuraamaan Istekin tarpeiden mukaisesti. Istekillä oli jo alun perin käytössä Torrus -niminen monitorointipalvelu, jota oli tarkotus lähteä kehittämään. Suurin monitoroinnin parannustoive oli yksinkertaistaa ja selkeyttää nykyistä mallia.

Työn alussa tehtiin paljon tutkimus- ja tutustumistyötä liittyen työhön. Juniperin osalta perehdyttiin sen reitittimiin ja kytkimiin, sekä verkoitustekniikoihin, kun taas Torruksen osalta yleisesti vain palveluun ja sen konfigurointiin. Tutkimus- ja tutustumistyö antoi hyvän pohjan opinnäytetyössä tehtyyn käytännön osuuteen, jonka avulla Monitorointi voitiin toteuttaa.

2 MULTIPROTOCOL LABEL SWITCHING

2.1 Yleistä

MPLS on siirtomekanismi, joka kuljettaa dataa pakettivälitteisen verkon kautta. Se on laajalti käytössä palveluntarjoajien ja suuryritysten verkostoissa. Lisäksi MPLS on suunniteltu tarjoamaan joustavuutta toimimaan lähes minkä tahansa Layer 3- ja Layer 2 – teknologian kanssa. MPLS-pohjaiset ratkaisut voidaan yhdistää saumattomasti olemassa oleviin infrastruktuureihin. (Bahaiji, 2016.)

Tällä tekniikalla parannellaan IP-reitityksen runkoverkkojen huonoja puolia (Kaario 2002, 126). Tällä Hetkellä MPLS määritellään vain IP: lle. Yhteyspohjaisen lähestymistavan avulla MPLS: n pakettien edelleenlähetys perustuu olemassa oleviin polkuihin. MPLS-välityspolkua voidaan myös ajatella tunneliksi, joka kulkee MPLS: n sisääntulopisteestä MPLS: n ulostulopisteeseen. Verrattuna perinteiseen IP-reititykseen ja välitykseen, polun valinta ja pakettien edelleenlähetys erotetaan MPLS: ssä. (Zhang ja Bartell, 2016.)

Ideana MPLS-verkossa on lisätä IP-pakettien eteen verkon reunalla tunniste (Label). Tunnisteen vuoksi paketin kulku on MPLS-verkossa ennalta määrätty ja siksi IP-liikenne kulkeekin MPLS-verkossa samalla tavoin, kuin kytkentäisen verkon sisällä. Lisänä tässä on siis tunnus, jonka avulla data voidaan tunnistaa ja jonka avulla se osataan nopeasti kytkeä etenemään oikeille reiteille. Myös vaihtoehtoisia reittejä on mahdollista määrittää ja niitä voi olla kullakin IP-paketilla. IP-paketin poistuessa MPLS-verkon sisältä, ylimääräiseksi jäävä otsikko otetaan pois, jonka takia MPLS-verkko kokonaisuudessaan näyttää IP-tasolla yhtenä reittittimien välisenä hyppynä. (Kaario 2002, 126-127)

2.2 Etuja ja hyötyjä

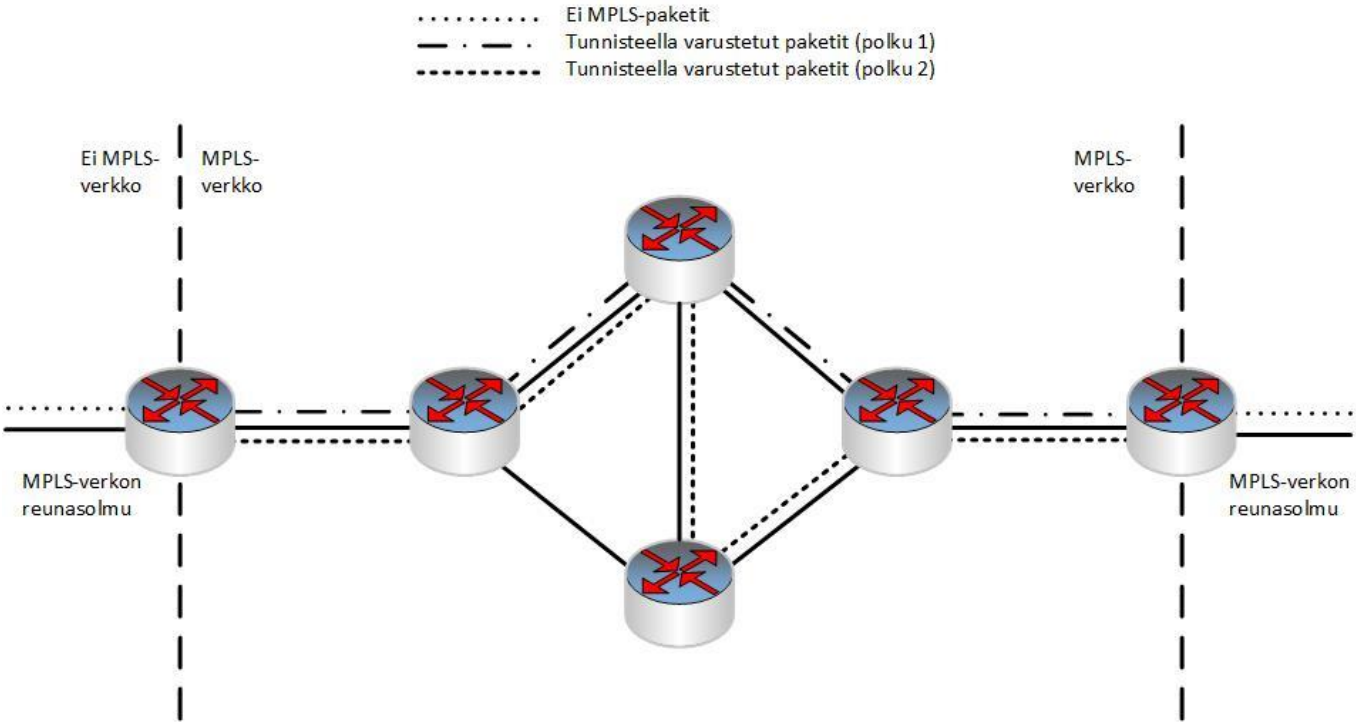
Oleellisimpia etuja MPLS-verkossa tai MPLS-reitityksessä on asioiden ja tehtävien minimointi verkon sisällä. Perinteisten reitittimien täytyy laskea jokaisella hypyllä, kun taas MPLS-verkossa suurin työ tehdään reittiä neuvoteltaessa verkon reunalla. Kun reitti on verkon reunalla neuvoteltu ja suunniteltu, paketin käsittely verkon sisällä on erittäin nopeaa (Kaario 2002, 127). Juuri tämän takia MPLS-verkon käsittely on paljon helpompaa verrattuna perinteiseen IP-reititykseen. Verkko on myös paljon nopeampi, joten se antaa käyttäjilleen paljon etuja ja mahdollisuuksia isompienkin pakettien kuljetukseen.

MPLS mahdollistaa myös puhtaaseen IP-reititykseen perustuvien verkkoihin verrattuna paremman verkon kuormanjaon sunnittelun. Reititetyn OSPF-verkon tapauksessa esimerkiksi reititysprotokolla tekee lyhimmän polun kahden IP-aliverkon välille ja peräkkäin liikkuvat IP-paketit kulkevat hyvin todennäköisesti myös samaa reittiä. Monessa tapauksessa "sivussa" sijaitsevat verkon osat saattavat olla alikuormitettuja samaan aikaan kun jokin toinen osa verkosta tai niin sanottu pullonkaulareititin on ylikuormitettuna. MPLS-verkossa tällaisissa tapauksissa voidaan luontevasti määritellä paikasta A paikkaan B useita vaihtoehtoisia reittejä. Tämä on perinteisin ATM-verkoista jonka takia ATM onkin vahvasti MPLS: n määrittelyjen taustalla. (Kaario 2002, 128)

2.3 Toimintaperiaate

Joukko MPLS-reitittimiä joita kutsutaan LSR-reitittimiksi (Label Switch Router) muodostavat MPLS-verkon. LSR-reitittimet vastaavat siitä, että MPLS-verkon reunalla kuhunkin pakettiin lisätään oikea tunniste (Label) kun ne tulevat verkkoon. Kun paketti taas poistuu verkosta, siitä samalla tavalla poistetaan tunniste LSR-reitittimien toimesta. Verkon LSR-Reitittimen täytyy ratkaista reitti (tai useampi vaihtoehtoinen reitti) lähteestä kohteeseen, jotta oikeaa tunnistetta voidaan käyttää. Jotta tunnistetietoja voidaan välittää MPLS-verkossa, tarvitaan siihen jotain jakeluprotokollaa. Tämänkaltaiset protokollat kulkevat yleisesti nimellä Label Distribution Protocol (LDP). (Kaario 2002, 128–129)

MPLS-tunniste on yksikäsitteinen vain kahden vierekkäisen LSR-laitteen välisellä linkillä. Tämä periaate on myös ATM-protokollassa täysin samanlainen. Jokaisella LSR-laitteella täytyy siis olla tunnistetaulu (label map), jonka avulla välittäessä dataa eteenpäin se voi suorittaa otsikon vaihdon. MPLS-yhteydelle voidaan myös määrittää vaihtoehtoisia polkuja, kuten alla olevassa kuvassa. (Kaario 2002, 129)



Kuva 1. MPLS: n vaihtoehtoiset polut.

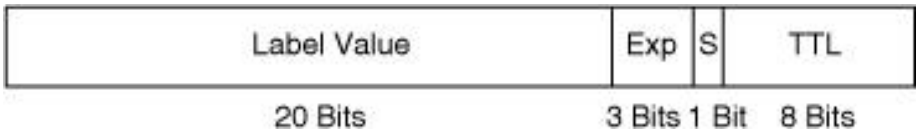
2.4 MPLS-Arkkitehtuuri

2.4.1 Tunniste ja tunnistepino

MPLS-tunnisteet voivat olla erilaisia riippuen taustalla olevasta linkistä. Tunnisteiden osalta on olemassa kolmea erilaista tyyppiä:

- Kehyspohjainen
- Solupohjainen
- Ei-paketti-pohjainen

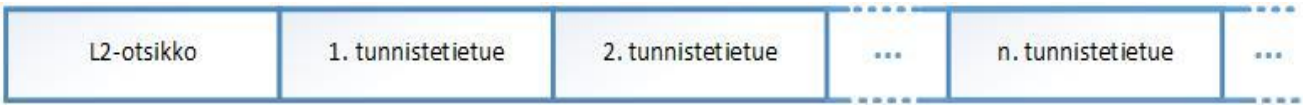
Alla olevassa kuvassa näkyy tunnisteen muoto kehys-pohjaisessa verkossa, kuten ethernet. Tunniste on kenttä, jolle on varattu 20-bittiä, jolla on tunnisteen todellinen arvo. Kun merkitty paketti on vastaanotettu, pinon yläosassa oleva tunnistearvo tarkistetaan (pino on yhden tai useamman tunnisteen ketju). Hakuoperaatiosta seuraa paketin seuraava hyppy ja toiminnan tyyppi voidaan määrittää tunnisteessa ennen lähetystä. (Zhang ja Bartell 2016.)



Kuva 2. Tunnisteen muoto kehys-pohjaisessa verkossa (Zhang ja Bartell 2016).

Exp-kenttä on 3 bittinen ns. Experimental Bits (Exp) käytetään tyypillisesti paketin palvelun luokan välittämiseen, niinkuin prioriteettibitit tekevät IPv4-ostsikossa. S-kenttä, Kun pinon (S) bitti on asetettu arvoon 1, nykyinen tunniste on pinon pohjalla. Koska MPLS-verkossa voi olla useita tunnisteita pinossa, tarvitaan erillinen pinon loppua merkitsevä bitti ja se on tämä. TTL-kenttä on 8 bittinen Time To Live (TTL) -kenttä ja sitä käytetään realiaikaisen arvon koodaamiseen. Jos merkityn paketin lähtevä TTL on 0, paketin verkon käyttöikä katsotaan vanhentuneen. Pakettia ei saa siirtään eteenpäin joko merkittynä tai merkitsemättöminä. (Zhang ja Bartell 2016.)

Tunnistepino (Label stack) on pino, jota MPLS-paketti voi siis kuljettaa mukanansa. Nimensä mukaisesti se on joukko tunnisteita. Esimerkiksi tunnelointiratkaisuja voidaan toteuttaa tunnistepinon avulla. Tunnelointi siis tarkoittaa yksikertaisesti sitä, että esimerkiksi ulkopuolisen palveluntarjoajan verkosta ohjataan joukko erilaisia yhteyksiä yhteiseen "tunneliin", kun taas omassa verkossa erotetaan ja puretaan tunneli alkuperäisiksi erillisiksi yhteyksiksi. MPLS-paketin tunnistepinossa voi olla yksi tai useampia tunnisteita ja niiden lukumäärä voi vaihdella paketin kulkiessa verkossa. MPLS-reitittimillä joilla tunnistepinoa käsitellään, on oleellista tunnitepinojen hallinnassa olevat säännöt. Tunnisteita voidaan poistaa tai lisätä sekä niitä voidaan vaihtaa joukoksi toisenlaisia tunnisteita. Tämä tarjoaa joustavan käsittelyn tunnisteisiin ja mahdollistaa monenlaisia sovelluksia. (Kaario 2002, 130)

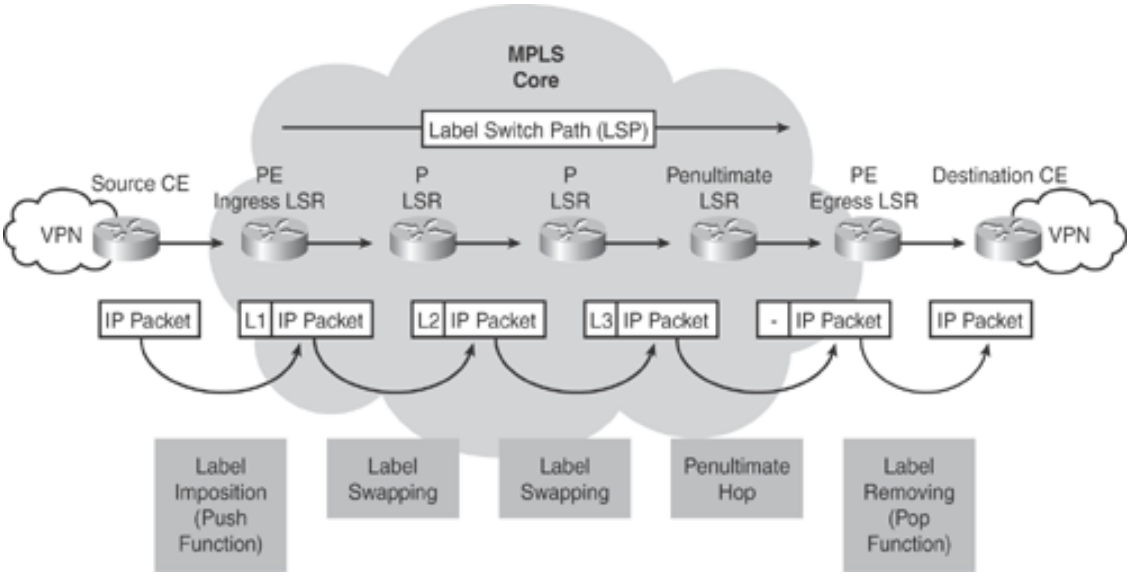


Kuva 3. MPLS-paketin tunnistepinossa voi olla lukuisa määrä tunnisteita.

2.4.2 LSR

LSR Eli Label Switch Router on MPLS-verkossa käytettävä reititin. Reitittimen toiminta perustuu siihen, että se vastaanottaa paketin ja lähettää sen MPLS-verkon lävitse. Paketin reitti määritellään jo reunareitittimellä, mikä tekee paketin matkasta sujuvaa ja nopeaa.

Kuten alla olevassa kuvassa näkyy, kukin LSR suorittaa tietyn function; Esimerkiksi reunassa oleva LSR suorittaa joko tunnisteen asettamisen tai tunnisteen poiston. Muut LSR: t reitillä yksinkertaisesti vaihtavat tunnisteita. (Bahaiji, 2016.)



Kuva 4. LSR-reittitimen toiminta MPLS-verkossa. (Bahaiji, 2016.)

Jokainen LSR ylläpitää Label Forwarding Information Base (LFIB) -taulukkoa, joka on rakennettu käyttämään IP reititys taulua tunnistamaan tunnisteen ”sitovaa vaihtoa”. LFIB tarjoaa saapuvan tunnisteella varustellun paketin lähtevälle liitännälle ja uuden tunnistetiedon, joka liittyy vastaavasti lähtevään pakettiin. (Bahaiji, 2016.)

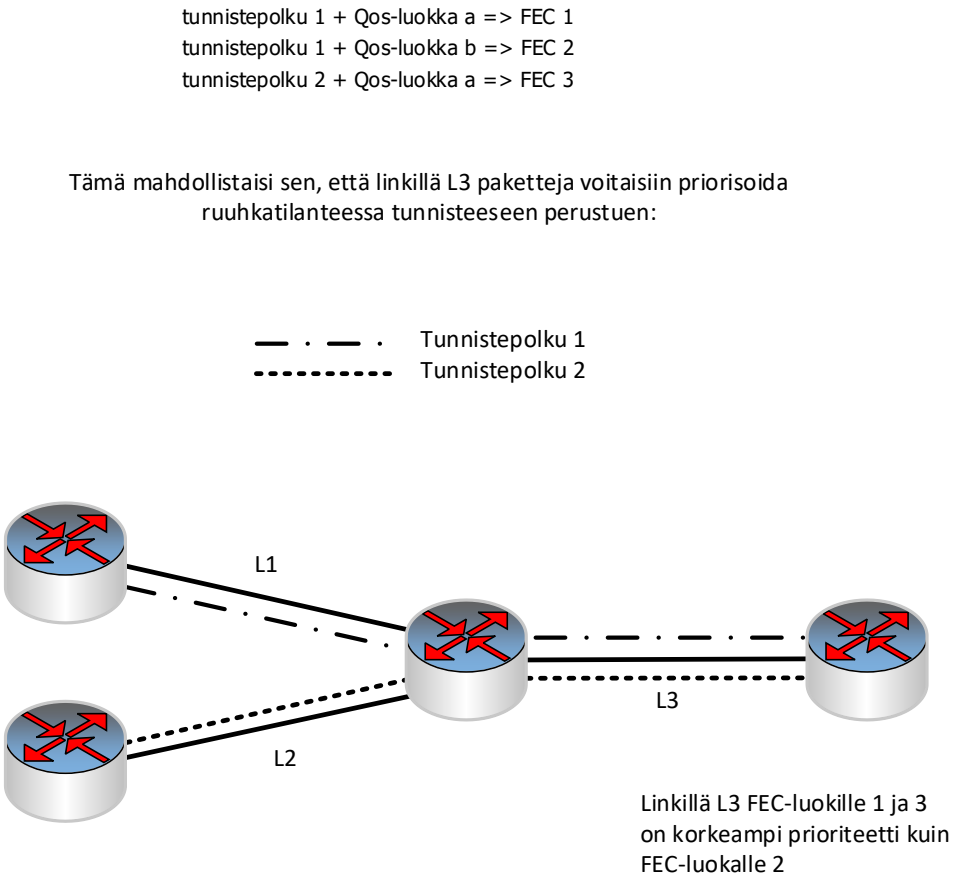
2.4.3 Tunnistepolku

LSR-reitittimestä matka toiseen LSR-reitittimeen MPLS-ytimen sisällä ylittää useita LSR-reitittimiä. Tätä polkua kutsutaan nimellä Label Switched Path (LSP). LSP-on olennaisesti joukko LSR: iä, joiden kautta leimattujen pakettien on mentävä päästäkseen LSR: ien reunalle. Koska paketti kulkee LSP: n läpi, jokainen LSR vaihtaa tunnisteen, kunnes se saavuttaa reitittimen ennen viimeistä LSR: a (viimeisimmän hopin), joka poistaa tunnisteen ja välittää paketin ilman tunnistetta viimeiselle LSR:lle, jossa paketti on pois MPLS-ytimestä ja lähetetään määränpäähän XY. (Bahaiji, 2016)

2.4.4 FEC

Jokaisella tunnistepolulla voi olla samaan aikaan menossa useita eri palvelunlaatuvaatimuksilla varusteltuja MPLS-paketteja. Silloin on normaalia, että niitä ei kohdella samoin tavoin ruuhkatilanteessa, vaikka itse reitti onkin sama. Mikäli kahdella eri paketilla on sama tunnistepolku ja samat QoS-vaatimukset, sanotaan niiden kuuluvan samaan FEC-luokkaan. FEC-luokat ovat MPLS-verkkon kytkentäsääntöjen perusta. (Kaario, 2002.)

Sama tunnistepolku ei kerro kahden MPLS-paketin kuuluvan samaan FEC-luokkaan, sillä QoS-vaatimukset voivat aiheuttaa tarvetta sijoittaa samalle polulle kuuluvat paketit eri FEC-luokkiin. Esimerkiksi alla olevassa kuvassa näkyvä määrittely. (Kaario, 2002.)



Kuva 5. FEC-tunnistepolut.

2.5 QoS

2.5.1 Yleistä

QoS on käsite tarjota eri palvelutasoja valittua liikennettä varten. Tällöin laitteille annetaan tietoja, joiden avulla määritellään, miten käsitellä laitteiden liikennettä, kun se on suurempi kuin käytettävissä olevat resurssit. Nämä resurssit voivat olla kaistanleveyttä, pääsyluetteloita, käsittelyä ja muita resursseja. (Noble, 2017.)

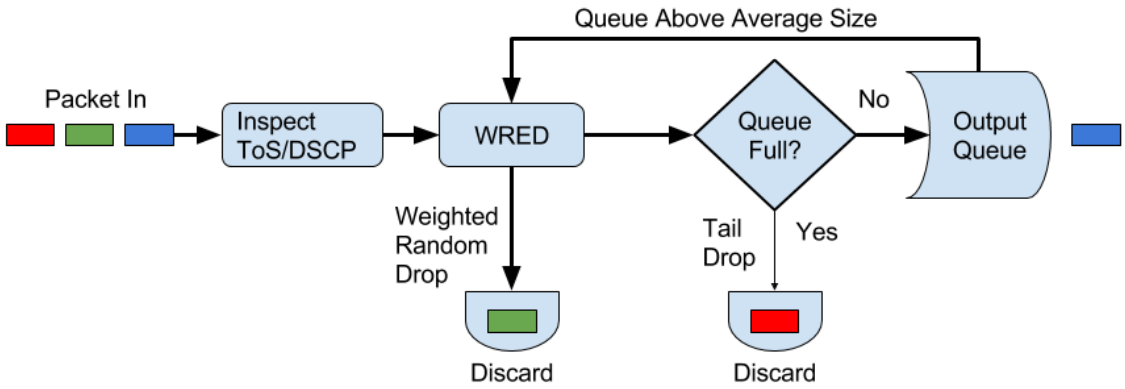
On tärkeää huomata, että vaikka liikenne ei näytä ylittävän linkkiä, mikro-burstit voivat aiheuttaa vaikeasti korjattavia ongelmia. Jopa QoS: n ja muiden liikenteenhallintatyökalujen avulla on aina suositeltavaa pitää linkin käyttö alle 80 prosentissa. (Noble, 2017.)

2.5.2 Toimintatavat

QoS: llä on kaksi päätapaa, joilla se voi hallita liikennettä; joko priorisoimalla tiettyjä virtauksia tai rajoittamalla kaikki muita virtoja. Normaalilaitteessa QoS: llä voit eriyttää palveluja mihin tahansa yhteensopivaan paketin otsikkotietoon, erimerkiksi lähde- tai kohdeosoitteeseen. Jotkin laitteet, jotka tekevät paketin syvällistä tarkastusta, voivat vastata lähes mihin tahansa paketissa. (Noble, 2017.)

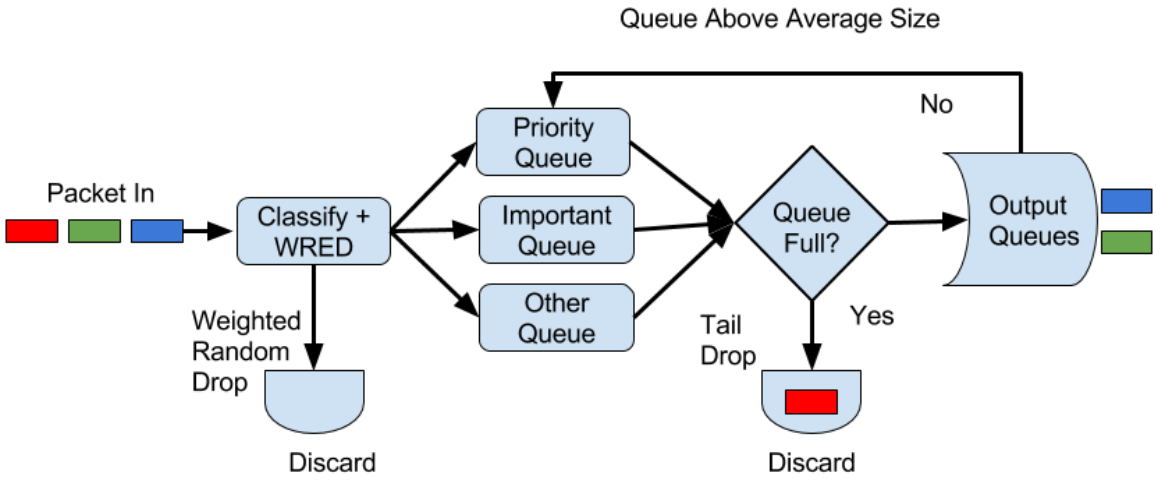
Kun QoS-protokollaa käytetään linkkinä, on monia tapoja hallita liikennettä joka ylittää verkon rajat, sisältäen tail dropin, Weighted round Robinin (WRR), Random Early Detectionin (RED) ja Weighted Random Early Detectionin (WRED). Kaikki nämä jononhallintaalgoritmit tekevät samoja asioita; poistaa tai hylkää paketit jonosta. (Noble, 2017.)

Tail drop on yksinkertaisin tapa hallita liikenneongelmia. Tail dropilla jonon tai muistitilan ollessa lopussa, paketit pudotetaan, kun ne yrittävät liittyä jonolle joko tulopuolella tai verkossa. WRR taas on algoritmi, joka alun perin sunniteltiin ATM linkeille, missä paketit ovat aina saman kokoisia. Käytettäessä linkkejä, joissa on satunnaisia pakettikokoja, on laskettava keskimääräinen paketin koko ja sovellettava sitä. WRR: tä ei ole nähty paljon verkoissa näinä päivinä ja siksi se on korvattu painotetulla Fair Queuing (WFQ) painolla. RED, joka tunnetaan myös nimellä Random Early Discard / Drop, on oikeidenmukaisempi jononhallinta-algoritmi kuin Tail drop. RED toimii tarkkailemalla linkin tulostusjonoa ja jonon ollessa täynnä, se alkaa satunnaisesti pudottaa paketteja, jotka yrittävät liittyä jonoksi; Jos jono tulee täysin täyteen, kaikki paketit pudotetaan. RED-peruskysymyksen tärkein seikka on se, että se ei huomio QoS-prioriteetteja. WRED algoritmi on muunnos RED: stä. Sen avulla jonoilla on useita kynnysarvoja liikenneluokan (ToS, Cos tai DSCP) perusteella. Kun paketit tulevat ja jos keksimääräinen jonon koko on pienempi kuin vähimmäiskynnys, paketit ovat jonossa. Kunj ono on täynnä, alemman prioriteetin paketit hylätään ennen korkean prioriteetin paketteja. (Noble, 2017.)



Kuva 6. WRED esimerkki kolmella erityyppisellä paketilla. (Noble, 2017)

Edellisessä kuvassa näkyy muutamia QoS-tekniikoita käytössä. Ensin tulee kolme pakettia: sininen, vihreä ja punainen. Kun sininen paketti menee läpi, WRED ei ole käytössä, joten sininen paketti siirretään output-jonoon. Sininen paketti siis päästettiin output-jonoon, koska jono ei ollut täynnä. Kun vihreä paketti lähtee, WRED otetaan käyttöön, mutta koska jono on korkeampi keskiarvoa, vihreä paketti hylätään. Punaisen paketin tullessa läpi, jono on keskimääräistä pienempi, joten paketti lähetetään jonoksi; Jono on täynnä, joten punainen paketti hylätään. Jos tähän lisättäisiin luokitus, eli pakettien ToS / DSCP -lukemisen, saamme jotain vastaavaa kuin alempana kuvassa. (Noble, 2017.)



Kuva 7. WRED paketti luokitus perustuen DSCP: hen. (Noble, 2017.)

Kuvassa 7 on kolme pakettityyppiä, joista jokaisella on erilainen DSCP-arvo: sininen on etusijalla, vihreä on tärkeä ja punainen kuuluu yleiseen jonoon. Siniset ja vihreät paketit läpäisevät ilman mitään ongelmia, mutta koska muut rivet ovat täynnä, punainen paketti hylätään. (Noble, 2017.)

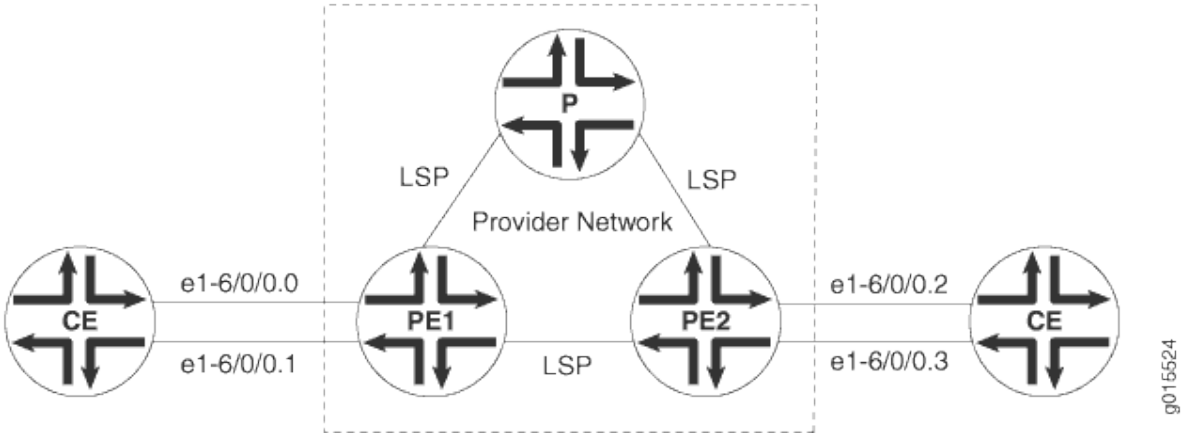
2.6 MPLS VPN

2.6.1 Yleistä

Virtual private networks (VPN) ovat yksityisiä verkkoja, jotka käyttävät julkista verkkoa kahden tai useamman etäisen sivuston muodostamiseksi. Verkkojen omien yhteyksien sijasta VPN-verkot käyttävät virtuaalisia yhteyksiä, jotka on reititetty (tunneloitu) julkisten verkkojen kautta, jotka ovat tyypillisesti palveluntarjoajaverkkoja. VPN-verkot ovat kustannustehokas vaihtoehto. VPN: n tyyppi on määritelty sen käyttämien yhteyksien mukaisesti ja että vaatiiko asiakkaan tai palveluntarjoajan verkko virtuaalista tunnelointia. (Juniper Networks, 2016.)

2.6.2 MPLS VPN Topologia

On monia tapoja pystyttää MPLS VPN ja suora liikenne sen kautta. Alla oleva kuva kertoo miltä näyttää tyypillinen MPLS VPN Topologia. (Juniper networks, 2016.)



Kuva 8. Tyypillinen MPLS VPN Topologia. (Juniper Networks, 2016.)

MPLS VPN -laitteita on kolme ensisijaista tyyppiä: Layer 2 VPN, Layer 2 piirit ja Layer 3 VPN. Kaikissa MPLS VPN -laiteissa on tiettyjä komponentteja. (Juniper Networks, 2016.)

Palveluntarjoajan reuna (PE) reitittimet palveluntarjoajan verkossa yhdistyvät asiakasreittien (CE) reitittimiin. PE-reitittimet tukevat VPN- ja MPLS-tunnisteiden toimintaa. Yksittäisen VPN-yhteyden sisällä PE-reitittimien parit kytketään virtuaalisen tunnelin kautta tyypillisesti etikettikytkentäiseen polkuun (LSP). (Juniper Networks, 2016.)

Palveluntarjoajan verkon ytimen sisällä olevat palveluntarjoajat eivät ole yhteydessä mihinkään reitittimeen asiasivustossa, vaan ovat osa PE-reitittimien parien tunnelia. Palvelun reitittimet tukevat LSP-toimintoja osana tunnelia, mutta eivät tue VPN-toimintoja. (Juniper Networks, 2016.)

CE-reittitimet ovat reittitimiä tai kytkimiä, jotka sijaitsevat asiakaspaikassa, joka muodostaa yhteyden palveluntarjoajan verkkoon. CE-reitittimet ovat tyypillisesti IP-reitittimiä, mutta ne voivat myös olla Asynchronous Transfer Mode (ATM), Frame Relay tai Ethernet-kytkimiä. (Juniper Networks, 2016.)

Kaikki VPN-toiminnot suoritetaan PE-reitittimillä. Jopa CE-reitittimien tai toimittajan reitittimien ei tarvitse suorittaa mitään VPN-toimintoja. (Juniper Networks, 2016.)

2.6.3 MPLS VPN Reititys

VPN-tunnelit liikennöidään yhdestä asiakaspaikasta toiseen asiakaspaikkaan käyttämällä julkista verkkoa kauttakulkuverkostona, kun tiettyjä vaatimuksia noudatetaan. Ensinnäkin liikenne välitetään CE-reitittimistä PE-reitittimiin tavallisella IP-välityksellä. Lisäksi PE-reitittimet muodostavat LSP: n palveluntarjoajan kautta. Saapuva PE-reititin vastaanottaa liikennettä ja suorittaa reittisuunnuksen. Haku tuotta LSP: n seuraavan hopin, ja liikenne välitetään LSP: lle. Lopuksi liikenne tavoittaa lähtevän PE-reitittimen ja PE-reititin poistaa MPLS-tunnisteen ja välittää liikenteen tavallisella IP-reitityksellä. (Juniper Networks, 2016.)

2.6.4 VRF

Reititysinstanssit ovat kokoelma reititystaulukoita, rajapintoja ja reititysprotokollaparametreja. Liitännät kuuluvat reititystaulukoihin ja reititysprotokollaparametrit ohjaavat tietoja reititystaulukoissa. MPLS VPN: n tapauksessa kullakin VPN: llä on VPN-reititys ja välitys VRF, joka tulee sanoista VPN routing and forwarding. (Juniper Networks, 2016.)

VRF-instanssi koostuu yhdestä tai useammasta reititystaulukosta, johdetusta välitystaulukosta, välityspöytiä käyttävistä rajapinnoista sekä käytännöistä ja reititysprotokollista, jotka määrittävät, mitä siirtotaulukkoon tulee. Koska jokainen instanssi on määritetty tietylle VPN: lle, jokaisella VPN: llä on erilliset taulukot, säännöt ja käytännöt, jotka ohjaavat sen toimintaa. (Juniper networks, 2016.)

Jokaiselle VPN: lle, joka on yhteydessä CE-reitittimeen, luodaan erillinen VRF-taulukko. VRF-taulukko täytetään VRF-instanssin yhdeydessä olevilla suoraan kytketyillä CE-sivustoilla vastaanotetuilla reiteillä ja toisilla reiteillä, jotka on vastaanotettu muista VPN: n PE-reitittimistä. (Juniper Networks, 2016.)

3 JUNIPER NETWORKS

3.1 Yleistä

Useimmat ovat kuulleet Ciscosta, mutta melko harvat ovat kuulleet Juniper Networksista. Kuten Cisco, Juniper Networks valmistaa erilaisia verkon tietoturvalaitteita ja-ohjelmistoja. Juniper tarjoaa myös monipuolisen valikoiman sertifikaatteja verkoittamiseen liittyvään tuotelinjaan. Kuten Cisco, Juniper Networks tarjoaa useita sertifikaatti tasoja ja erilaisia kappaleita. Sen sertifikaatit auttavat henkilöstöä, joka työskentelee organisaatiossa, jossa käytetään Juniper Networks -laitteita, jotta käyttäjät saavat niistä kaiken irti. (Kim & Salomon, 2016.)

Sertifikaattihakijat voivat suorittaa kursseja ja tenttejä, jotta he voivat saada sertifikaatit neljällä eri tasolla 11 eri kappaleelta. Juniper Networks ei tarjoa sertifikaatteja kaikilla tasoilla jokaiselle kappaleelle. (Kim & Salomon, 2016.)

3.2 Junos OS

Junos-käyttöjärjestelmä on tarkoitukseen rakennettu verkko-operaatiojärjestelmä, joka perustuu yhteen maailman vakaimmista ja turvallimmista käyttöjärjestelmistä: FreeBSD. Junos-ohjelmisto on suunniteltu monoliittiseksi ytimen arkkitehtuuriksi, joka sijoittaa kaikki käyttöjärjestelmän palvelut ytimen tilaan. Junosin tärkeimmät komponentit on kirjoitettu daemoneina, jotka tarjoavat täydellisen prosessin ja muistin erottamisen. Junoksesta 14.x otettiin käyttöön suurin muutos – modulaarisuus. Vaikka Junos perustuu edelleen FreeBSD: ään, siitä tulee riippumaton ”guest OS”: sta ja se tarjoaa eron Core-käyttöjärjestelmän ja HW-ohjainten välillä. Lisäksi monia parannuksia on tulossa lähivuosina. (Hanks, Reynolds ja Roy. 2016.)

Kaikki Juniper-laitteet on siis suunniteltu FreeBSD-koodipohjan ympärille, mutta ihan kaikki Juperlaitteet eivät käytä FreeBSD-käyttöjärjestelmää. Sen sijaan ne käyttävät ydintä kehyksenä toiselle Junos-käyttöjärjestelmälle tarkoitetulle käyttöjärjestelmälle. Kuintekin koska Junos-käyttöjärjestelmä on suunniteltu FreeBSD: n ympärille, voit käyttää erityistä versiota tietokoneesi ohjelmistosta nimeltä Juniper Olive. (Neumann, 2015.)

Juniper Olive on Junosin käyttöjärjestelmän versio. Koska se toimii tavallisella tietokoneella eikä varsinaisella Juniper-alustalla, sillä ei ole omaa ASIC-laitteistoa, jolla saavutetaan todellinen Juniperlaitteen suuri läpäisykyky. Muuten se on todella toimiva ratkaisu, eikä mikään simulaatio käyttöjärjestelmästä. Tämän ansiosta se soveltuu hyvin Juniper Networks Certified Associate (JNCIA) -sertifikaation ja muiden aloitusstandardien sertifiointiin. (Neumann, 2015.)

Junosin käyttöjärjestelmä tulee uudistumaan vielä paljon. Skaalaustarkoituksessa se tarkoittaa sitä, että se tulee olemaan modulaarisempi, nopeampi ja helpompi tukea kaikkia uusia virtuaaliominaisuuksia, jotka tulevat SDN: n kannoille. Jo nyt Junos on siirtynyt viimeisimpiin ohjelmistorakenteisiin, kuten Kernel SMP ja multi-core-käyttöjärjestelmään. (Hanks, Reynolds ja Roy. 2016.)

Juniper tarjoaa myös Python-kirjaston nimeltä PyEZ Junos-käyttöjärjestelmälle sekä Windows järjestelmänvalvojille PowerShell-asetukselle, joka käyttää PowerShelliä Python-”kääröllä”. Python-kirjasto PyEZ voi hakea mitä tahansa konfigurointi informaatioita käyttämällä taulukoita ja näkymiä. Kun taulukko-osiot on otettu hyödyntämällä python-komentosarjaa käyttäen get () menetelmää, taulukoita voidaan myöhemmin käsitellä python-sanakirjana ja iteroituna, jonka avulla käyttäjät voivat suorittaa monimutkaisia komentosarjoja, mikä mahdollistaa verkko-operaattoreiden automatisoinnin kaikkiin verkkotoimintoihin. Junos PyEZ-kirjaston on myös täysin laajennettavissa ja verkko-operaattorit voivat lisätä toimintoja, joita he pitävät aiheellisina widget-järjestelmänsä avulla. (Armstrong, 2016.)

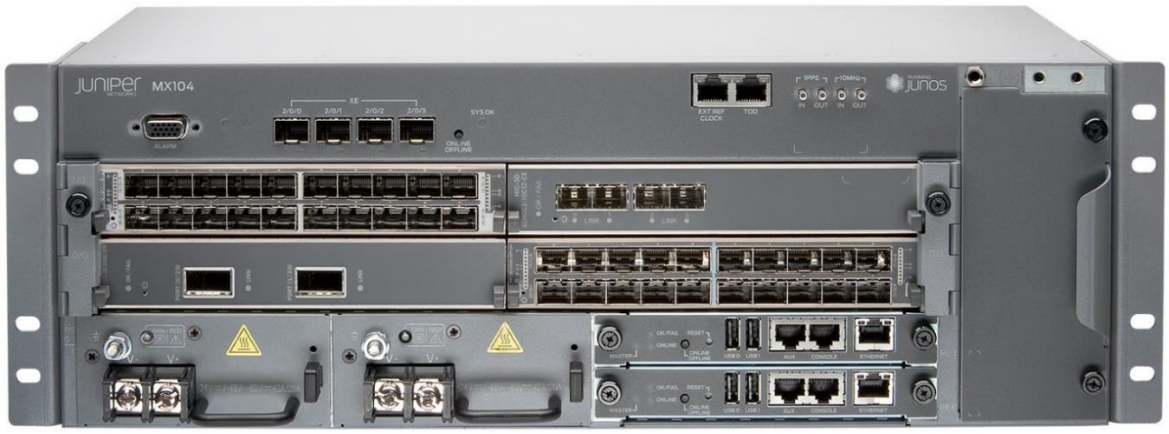
3.3 Reitittimet ja kytkimet

Juniperillä on käytössä monipuolisesti paljon erilaisia kytkimiä ja reitittimiä. Opinnäytetyön suorittamiseen käytin kuintekin vain yhtä reititinmallia ja yhtä pientä kytkintä. Tämän johdosta ajattelin, että tärkeintä olisi kertoa nimenomaan näistä laitteista, jotta opinnäytetyön laboratorion käytännön osuus tulisi selkeämmäksi.

3.3.1 MX104 Reititin

SDN-valmis MX104 3D Universal Edge -reititin on modulaarinan ja erittäin tarpeellinen ja monipuolinen MX-sarjan alusta, joka on rakennettu tila- ja teho-rajoitteisille tarjoajille sekä yrityksille. MX104 tarjoaa 160 Gigabittisen kapasiteetin, redundanttisen ohjaustason korkeaan käytettävyyteen, sekä neljä kiinteää 10GbE -porttia ja nelja modulaarista liitäntäkorttia, jotka mahdollistavat joustavan verkkoyhteyden sekä joustavia virtualisoituja verkkopalveluja. (Juniper Networks, 2016.)

Reititin käyttää Junos OS: ää ja ohjelmoitavaa Trio-piirisarjaa ja näin ollen MX104 jakaa samat edistyksdelliset reititys-, kytkentä-, turvallisuus-, ja huoltotoiminnot, jotka ovat käytettävissä suurissa MX-sarjan alustoissa, mukaan lukien tuki monille L2 / L3 VPN – palveluille ja kehittyneelle laajakaistaverkon yhdyskäytävätoiminnoille. MX104 auttaa verkko-operaattoreja muuttamaan verkostojaan ja yritystään menestyäkseen hyper-yhteysmaailmassamme. (Juniper Networks, 2016.)



Kuva 9. MX104 Reititin. (Juniper Networks, 2016.)

3.3.2 EX2200 ja EX2200-C kytkimet

EX2200 ja EX2200-C -kytkin on pienitehoinen ja pienikokoinen 1 U-laite, joka on suunniteltu kaapelointi kaappeihin, joten ne ovat edullinen ja helppo ratkaisu pienitehoisten liikenneyhteyksien tarpeisiin. EX2200-C: ssa ei ole tuuletinta, joten sen toiminta on erittäinhiljaista. Tämän takia kyseinen kytkin soveltuu hyvin myös esimerkiksi avoimii toimistomalleihin. (Juniper Networks, 2016.)

EX2200 tarjoaa 24 tai 48 10/100 / 1000BASE-T-porttimallia; EX2200-C tarjoaa 12 porttia. Molemmissa mallissa on joko tai 802.3af Power over Ethernet tai 802.3a PoE + -liitäntä tai 802.3at PoE + verkkolaitteiden tukemiseen. (Juniper Networks, 2016.)

Molemmat laitteet ovat skaalautuvia ja niitä voidaan käyttää Juniperin Virtual Chassis – teknologialla, joka yhdistää jopa neljä EX2200- tai EX2200-C-kytkintä yhdeksi loogiseksi laitteeksi. Kytkimet kytkeytyvät helposti runkoverkkoon etupaneelin Gigabit Ethernet -liitäntäporttien kautta. (Juniper Networks, 2016.)

Kuva 10. EX2200-C-kytkin, jota käytettiin opinnäytetyössä. (Juniper Networks, 2016.)

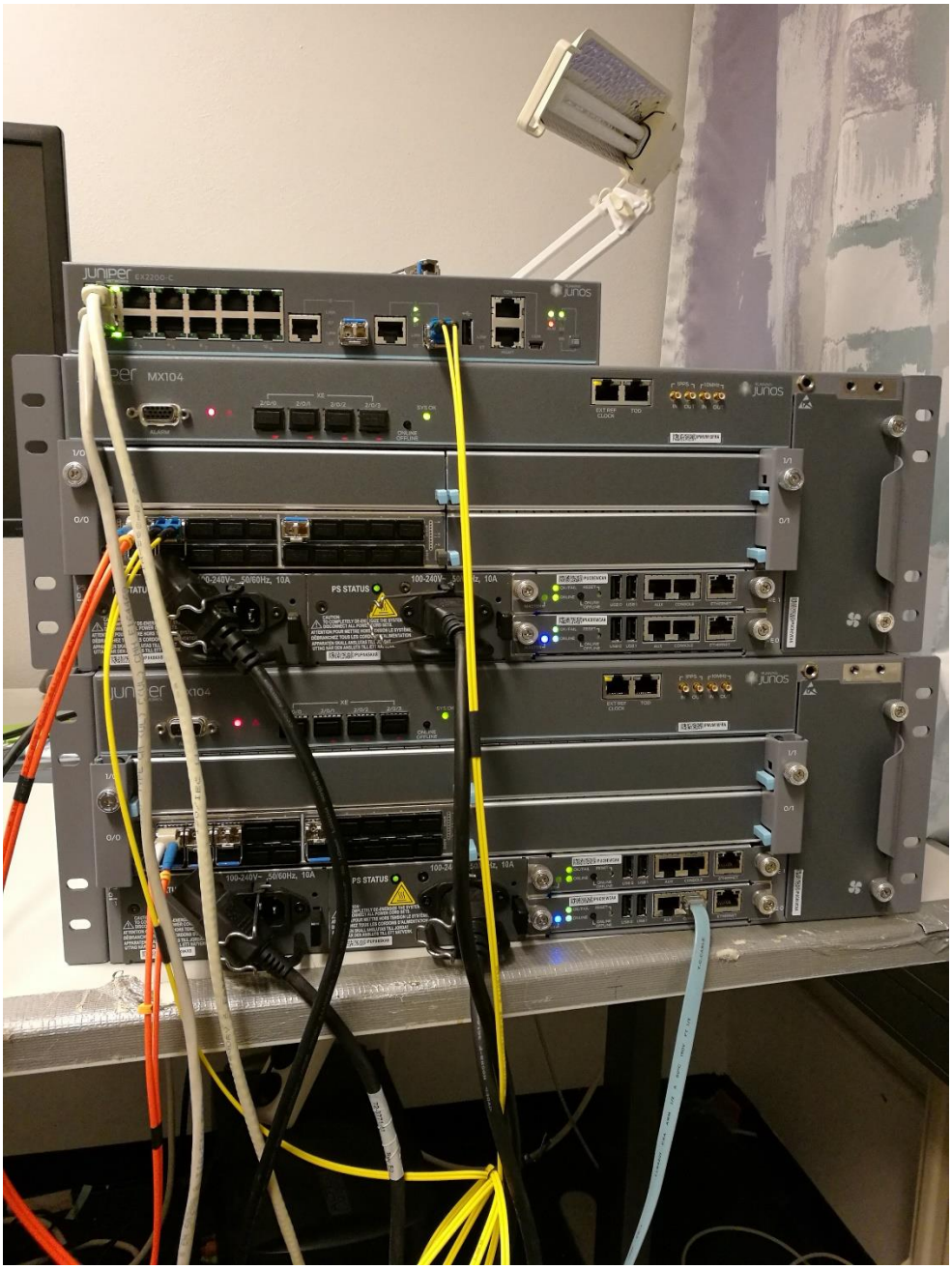
Kuva 11. EX2200-kytkin. (Juniper Networks, 2016.)

4 KÄYTÄNNÖN TESTIT JA TOTEUTUS

4.1 Testilaboratorion rakennus

Testilaboratorion rakentaminen aloitettiin kartoittamalla yleisesti tarvittavat laitteet sekä kaapelit. Tarvikkeet testilaboratorioon tarjosi Savonia ammattikorkeakoulu, lisäksi tilat testilaboratorion tekoon löytyivät myös sieltä.

Kun testilaboratorio oli saatu siihen vaiheeseen, että laitteita pääsisi konfiguroimaan, alkoi perehtyminen itse laitteisiin ja laboratorion rakentamiseen. Testilaboratorioon kuului Ubuntu-serveri, Windows pohjainen kannettava tietokone, kaksi Juniper MX104 – reititintä ja yksi Juniper EX2200-C – kytkin. Kummatkin reitittimet olivat kytkettynä toisiinsa kuitu-kaapelilla, ja toisesta reitittimestä lähti toinen kuitu-kaapeli kytkimelle. Kytkimeltä taas lähti kaksi Ethernet-kaapelia, toinen Windows koneelle ja toinen Ubuntu-serverille.

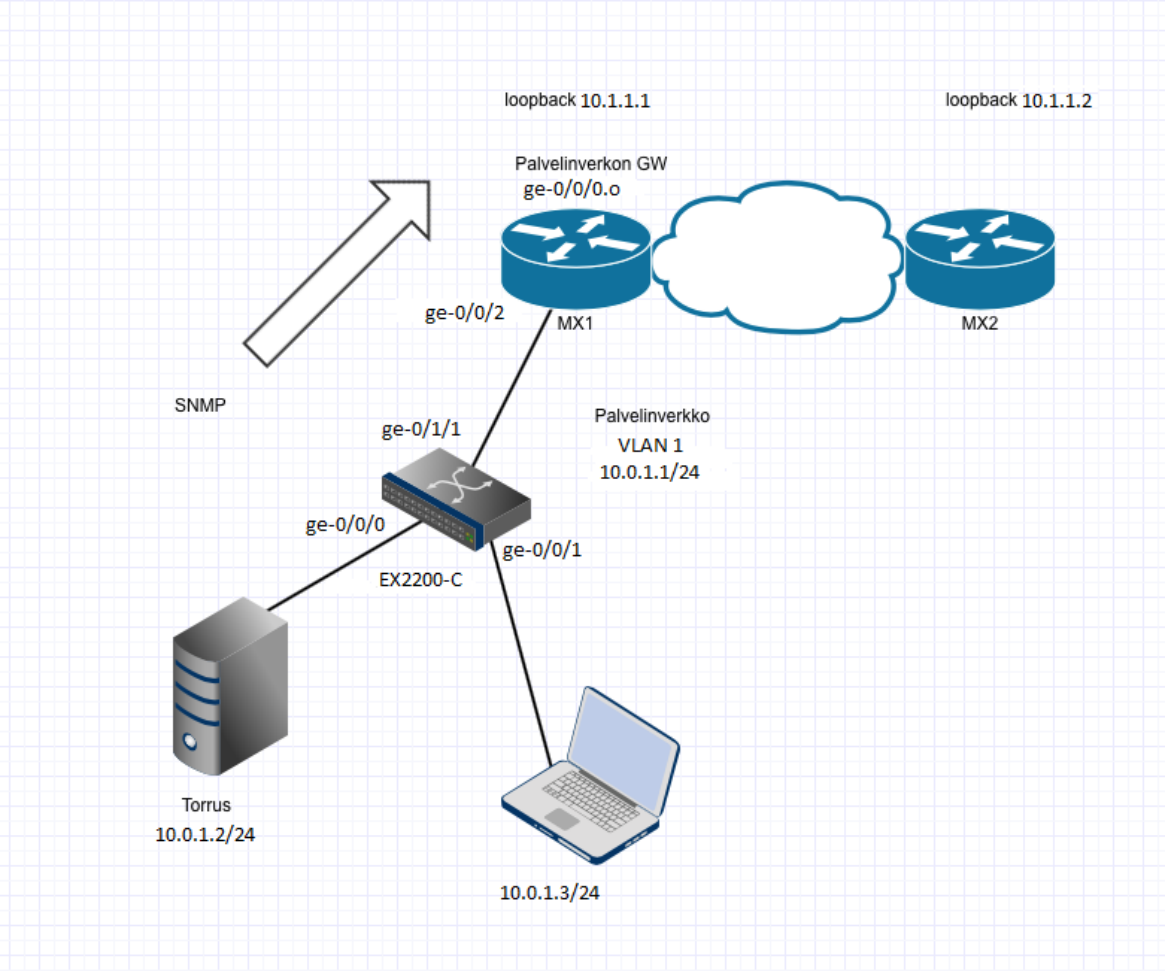


Kuva 12. Testilaboratorion verkkolaitteet.

Kuvassa 12 näkyy testilaboratoriossa käytetyt Reitittimet sekä kytkin. Kytkin päällimmäisenä ja reitittimet pohjalla. Oranssi kaapeli yhdistää reitittimet toisiinsa ja keltainen kaapeli yhdistää reitittimet kytkimelle, jonka kautta valkoisilla kaapeleilla jatketaan Windows koneelle ja Ubuntu-serverille.

4.2 Verkon konfigurointi

Laboratorioverkon konfigurointiin kuuluu Juniper MX104 – reitittimien ja Juniper EX2200-C - kytkimen konfigurointi. Lisäksi työn loppuvaiheessa tehtiin pieniä konfigurointeja myös windowsin puolella. Reitittimillä rakennettiin MPLS-verkko, jolla voitiin demota Istekin omaa verkkoa ja jälkeenpäin QoS-monitorointia käytännössä. Itse verkon rakennus oli erittäin haastava ja työläin vaihe, mutta se vaadittiin, jotta monitorointia päästiin testaamaan käytännössä.



Kuva 13. Verkon Topologia.

4.3 Ubuntu server

4.3.1 Yleistä

Ubuntu on käyttöjärjestelmä, joka perustuu Linux-kerneliin. Sen on luonut, kehittänyt, parannellut ja jakanut Ubuntu-yhteisö. Ubuntua sponsoroi Canoniacal Ltd. joka on avoimen lähdekoodin projektin tukema maailmanlaajuinen ohjelmistokehittäjäyhteisö. (Helmke, 2016.)

Ubuntu on uksi uusimmista Linux-jakeluista, jotka ovat tällä hetkellä saatavilla. Sen ensimmäinen versio julkaistiin lokakuussa 2004 ja se sai nopeasti maineensa asennuksen ja käytön helppouden vuoksi. Kuitenkin Ubuntu itse perustuu Debianiin, joka on paljon vanhempi jakelu, joka ulottuu laajemmasta Linux-yhteisöstä. (Helmke, 2016.)

Canoniacal-ohjelmiston sponsoroima ja Mark Shuttleworthin voimakkaiden resurssien avulla Ubuntu sai loistavan aloituksen Warty Warthog -versiolla 4.10. Alusta alkaen Ubuntu määritteli selkeät tavoitteet: tarjota jakelu, joka oli helppo asentaa ja käyttää, joka ei liioin sekoittanut käyttäjää ja joka tuli yhdelle CD-levylle (Nykyisin yksi DVD-kuva). Ubuntu julkaisi 6 uukauden välein nopean etenemisen Linux-yhteisöön ja on nyt yksi suosituimmista Linux-levyistä ympäri maailmaa. (Helmke, 2016.)

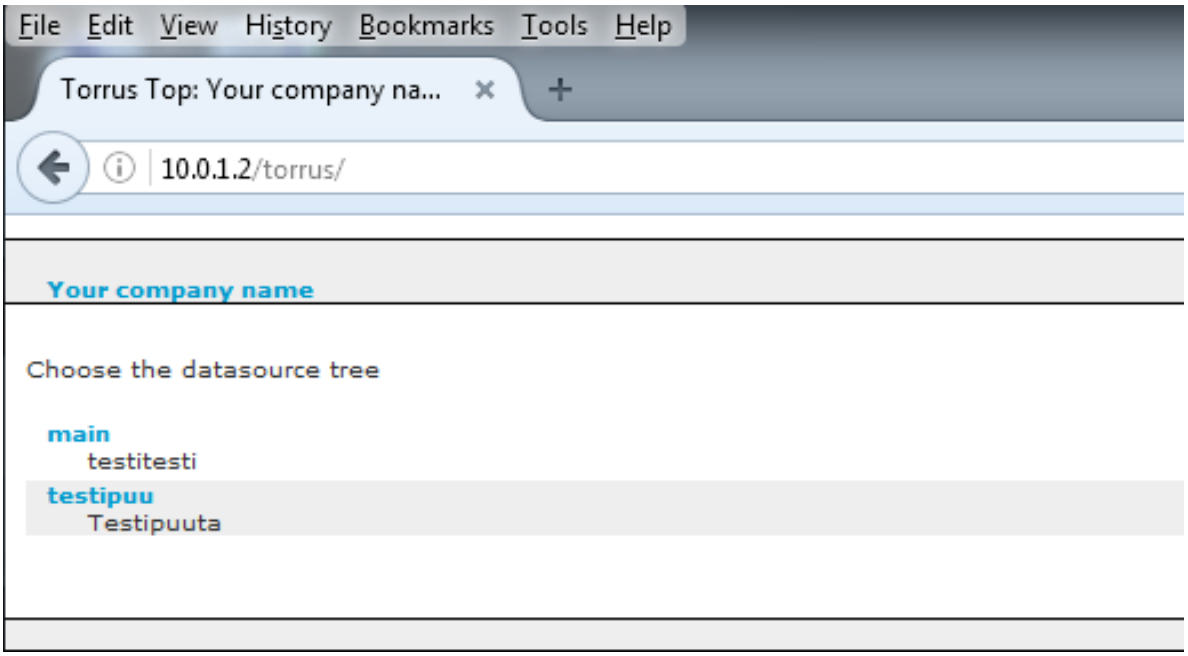
4.3.2 Torrus ja sen konfigurointi

Lyhyesti sanottanu Torrus on vaihtoehtoinen ohjelmistoalusta MRTGllä, Crichetillä ja Cactilla. Useimmissa tapauksissa se tuo lisää joustavuutta ja suorituskykyä. Torrus keräämään miljoona SNMP OID-tunnusta joka 5. minuutti modernista palvelimesta. (Torrus, 2017.)

Torrus on suunniteltu yleiseksi tietosarjojen käsittelyjärjestelmäksi. Sen skaalautuva hierarkkinen muotoilu, sovelluksen itsenäinen ydin, inrementaalinen konfigurointi ja erittäin muokattavissa oleva arkkitehtuuri tekevät Torrusesta houkuttelevan vaihtoehdon sekä pienille asennuksille että suurille yritys- tai operaattoriverkostoille. Vaikka suurin osa käyttäjistä käyttää Torrusia SNMP-valvontaan, se saattaa olla hyödyllistä kaikenlaisille tietosarjoille. (Torrus, 2017.)

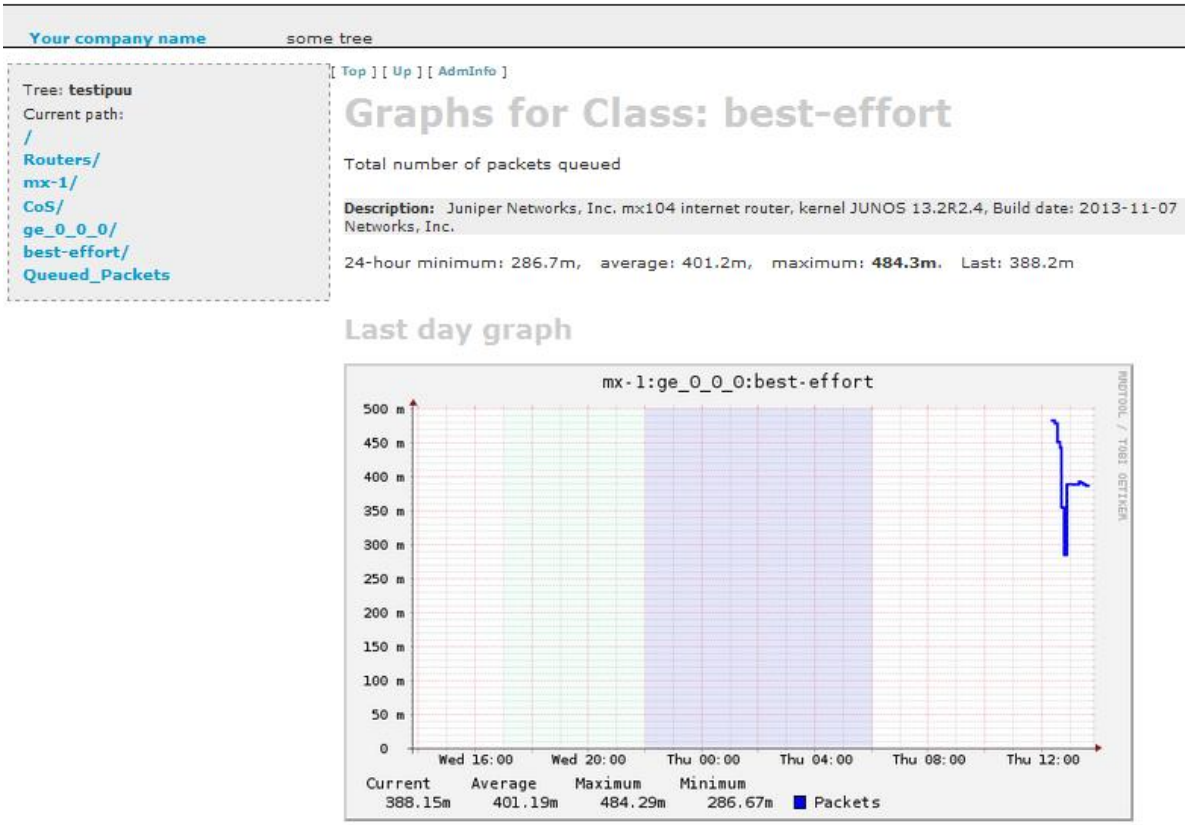
Torruksen konfigurointi oli todella iso osa työtä, ja vaati paljon perehtymistä. Jotta torruksen kongurointi voitiin aloittaa, piti itse Ubuntu-serveri konfiguroida toimivaksi ja määritellä sinne asetukset, jotta torrus toimisi. Alkutekijöissä asensin Ubuntun versio 16.10, mutta se paljastuikin vääräksi valinnaksi tukien kannalta. Ubuntu piti siis asentaa muutamaan otteeseen uudelleen, ja viimeinen versio, joka oli opinnäytetyön kannalta järkevin, oli Ubuntu 16.04.

Kun Ubuntu ja Torrus olivat molemmat kunnossa asennuksien suhteen, päästiin käsiksi torruksen käyttöliittymään, jonka kautta QoS-monitorointia seurataan. Itse monitorointia seurataan siis Windows koneen kautta siten, että kirjoitetaan koneen selaimeen Ubuntu-serverin osoite. Torrusta konfiguroidaan ja muokataan Ubuntussa, mutta sitä käytetään Windowsissa. Windows koneelle piti siis myös määrittää saman verkon osoite kuin Torrukselle, jotta se toimisi.



Kuva 14. Torruksen aloitussivu.

Kuvassa 13 näkyy Torruksen aloitussivu, johon päästään kirjautumis-ikkunan jälkeen. Aloitussivu on täysin muokattavissa, oikeastaan kaikin tavoin. Sinne voi vaihtaa fontteja, lisätä tekstiä, muokata tekstiä, lisätä linkkejä, muokata ulkoasua ja kaikkea muuta. Työssä pääasiallisesti oli ideana tehdä Torruksesta selkeämpi Istekin käyttöön. Jos klikkaa esimerkiksi ”testipuu” -linkkiä, pääsee testiverkossa olevien reitittimien tietoihin käsiksi, koska ne ovat konfiguroituna Torruksen kautta kyseiseen linkkiin. Torruksessa on vain se ongelma, että se kerää kaiken tiedon reitittimeltä, eikä sitä voi määritellä missään, mitä tietoa tulee ja mitä ei. Työssä oli tarkoitus saada tätä rakennetta ja tietoja vähemmälle sekä selkeämmäksi, mutta totesinkin työn aikana, että se ei yksinkertaisesti toimi.



Kuva 15. Torruksen QoS-monitorointi ikkuna.

Kuten kuvan 14 vasemmasta yläreunasta näkee, polku QoS-monitorointi ikkunaan on pitkä. Pääsivulta lähdettäessä täytyy mennä: ”testipuu/Routers/mx-1/CoS/ge_0_0_0/bets-effort/Queued_Packets”. Nimenomaan tätä rakennetta oltaisiin haluttu karsia, ja turhia vaihtoehtoja sulkea pois. Kun tämän vaihtoehdon muokkaaminen oli pois suljettu, täytyi lähteä kehittämään jotain muuta parannusideaa.

Päädyin lopulta ratkaisuun, joka parantaa huomattavasti nykyistä tilannetta. Ratkaisu liittyi etusivun konfigurointiin. Etusivun konfiguraatioita muokkaamalla saadaan ulkonäöstä sen näköinen kuin halutaan, ja jos laitteita halutaan eritellä esim. Kaupungeittain, onnistuu se nimeämällä ”puu” Esimerkiksi Kuopioksi, ja sen alta löytyisi kaikki Kuopiossa olevat laitteet. Jokaiselle puulle määritellään erikseen mitä laitteita sinne tulee, joten tällä tekniikalla reitittimien ja laitteiden erottelu esimerkiksi kaupunkien mukaan onnistuisi.

```
# Torrus Site config. Put all your site specifics here.
# You need to stop and start Apache server every time you change this file.

use Sys::Hostname qw(hostname);

@Torrus::Global::xmlAlwaysIncludeFirst = ( 'defaults.xml', 'site-global.xml' );

%Torrus::Global::treeConfig =
    (
     'main' => {
         'description' => 'testitesti',
         'info'        => 'some tree',
         'xmlfiles' => [qw(routers.xml)],
         'run' => { 'collector' => 1, 'monitor' => 0 } },
     'testipuu' => {
         'description' => 'Testipuuta',
         'info'        => 'some tree',
         'xmlfiles' => [qw(myrouters.xml)],
         'run' => { 'collector' => 1, 'monitor' => 0 } }
     );


# Customizable look in the HTML page top
# $Torrus::Renderer::companyName = 'Your company name';
# $Torrus::Renderer::companyURL = 'http://torrus.sf.net';
# $Torrus::Renderer::siteInfo = hostname();


1;
```

Kuva 16. Torruksen etusivun konfiguraatio.

Etusivua voitiin muokata kuvan 16 mukaisesta konfiguraatiosta Ubuntun puolella. Muokkaaminen oli suhteellisen yksinkertaista ja nopeaa. Konfiguraatioon pystyttiin lisätä lisää puita kopioimalla entisen konfiguraation ja lisäämällä sen perään. Puukonfiguraatiot eroteltiin pilkulla, kuten kuvassa näkyy.

```
     'testipuu' => {
         'description' => 'Testipuuta',
         'info'        => 'some tree',
         'xmlfiles' => [qw(myrouters.xml)],
         'run' => { 'collector' => 1, 'monitor' => 0 } }
```

Kuva 17. Yhden puun konfigurointi.

Yhden puun konfigurointi oli helppoa. Ylin ’testipuu’ -kenttä on siis linkin nimi, joka näkyy pääsivulla. Jos sen haluaisi muuttaa esimerkiksi Kuopion puuksi, laitettaisiin sen tilalle vain ’Kuopio’. Description -kenttään voitiin kuvata kuvaus, joka näkyi pääsivulla linkin alla. Info -kenttään kuvattiin teksti, joka näkyi yläpalkissa sittemmin, kun pääsivun linkkiä oli klikattu.

5 YHTEENVETO

Työn tarkoitus oli kehittää Istekille MPLS-runkoverkon monitorontikeino. Työ vaati paljon perehtymistä siihen liittyviin laitteisiin ja tarvikkeisiin. Työn tarkoitus oli myös oppia käyttämään Juniperin laitteita ja päästä tutkimaan käytännössä miltä QoS-jonojen monitorointi näyttää.

MPLS-verkon rakentaminen ja QoS-monitorointi ovat aiheena todella suuri osa nykyaikaista tietoliikennetekniikkaa, sillä MPLS-runkoverkkoa käytetään monissa isoissa yrityksissä. Jatkoa ajatellen MPLS-verkko tule korvaamaan normaalin IP-pohjaisen reitityksen täysin.

Työn teon aikana opin erittäin paljon Juniperin laitteista, niiden konfiguroinnista ja käytöstä. Se antaa erittäin hyvät jatkomahdollisuudet työelämässä, sillä Juniper lisääntyy koko ajan kaikkialla. Myös Ubuntu tuli paljon tutummaksi kuin ennen. Työn aikana kohdattiin paljon ongelmia, mutta ne saatiin ratkaistua. Verkon luonti oli omasta mielestäni vaikein ja työläin vaihe. Lopulta kuinteki kaikki saatiin toimimaan niin kuin pitikin, ja työhön päästiin kunnolla käsiksi.

Työ oli haastava mutta samalla myös mielenkiintoinen. Rajaus onnistui erittäin hyvin ja sen takia tavoitteisiin päästiin hyvin. Monitoroinnin kehitystä tullaan jatkamaan ja kehittämään Istekin sisäisesti jatkossa.

6 LÄHTEET

Kaario, K, 2002. TCP/IP -verkot. Jyväskylä: Docendo Finland Oy.

Zhang, R. Bartell, M. (toim.) 2016. BGP Desing and Implementation, Cisco Press.

Bahaiji, Y, 2016. CCIE Professional Development Series Network Security Technologies and Solutions, Cisco Press.

Juniper Networks. 2016. Tech Library. [Viitattu 2017-8-5]. Saatavissa: https://www.juniper.net/documentation/en_US/junos/topics/concept/mpls-security-vpn-overview.html

Noble, S, 2017. Building Modern Networks, Packt Publishing.

Hanks, D.R., Reynolds, H., Roy, D. (toim.) 2016. Juniper MX Series, 2nd Edition, O'Reilly Media, Inc.

Armstrong, S. 2016. DevOps for Networking, Packt Publishing.

Kim, D. Solomon, M. (toim.) 2016. Fundamentals of Information Systems Security, 3rd Edition, Jones & Bartlett Learning.

Neumann, J, 2015. The Book of GNS3, No Starch Press.

Juniper Networks. 2016. Products-services. [Viitattu 2017-10-22] Saatavissa: https://www.juniper.net/us/en/products-services/routing/mx-series/mx104/

Junper Networks. 2016. Image Library. [Viitatattu 2017-10-23] Saatavissa: https://www.juniper.net/assets/img/products/image-library/ex-series/ex2200-c/ex2200-c-frontwtop-high.jpg

Helmke, M. 2016. Ubuntu Unleashed 2017 Edition, Sams.

Torrus. 2017. [Viitattu 2017-11-07] Saatavissa: http://torrus.org